TITLE OF THE INVENTION:

[0001] HARDWARE ASSIST FOR ADDRESS LEARNING

REFERENCE TO RELATED APPLICATIONS:

[0002] This application claims priority to United States Provisional Patent Application Serial No. 60/220,805 filed on July 25, 2000. The contents of the provisional application identified above is hereby incorporated by reference.

BACKGROUND OF THE INVENTION:

Field of the Invention:

[0003] The invention relates to a method and apparatus for high performance switching in local area communications networks such as token ring, ATM, ethernet, fast ethernet, and gigabit ethernet environments, generally known as LANs. In particular, the invention relates to a new switching architecture in an integrated, modular, single chip solution, which can be implemented on a semiconductor substrate such as a silicon chip.

Description of the Related Art:

[0004] As computer performance has increased in recent years, the demands on computer networks has significantly increased; faster computer processors and higher memory capabilities need networks with high bandwidth capabilities to enable high speed transfer of significant amounts of data. The well-known ethernet technology, which is based upon numerous IEEE ethernet standards, is one example of computer networking technology which has been able to be modified and improved to remain a viable computing technology. A more complete discussion of prior art networking systems can be found, for example, in SWITCHED AND FAST ETHERNET, by Breyer and Riley (Ziff-Davis, 1996), and numerous IEEE publications relating to IEEE 802 standards. Based upon the Open Systems Interconnect (OSI) 7-layer reference model, network capabilities have grown through the development of repeaters, bridges, routers, and, more recently, "switches", which operate with various types of communication media. Thickwire, thinwire, twisted pair, and optical fiber are examples of media which has been used for computer networks. Switches, as they relate to computer networking and to ethernet, are hardware-based devices which control the flow of data packets or cells based upon destination address information which is available in each packet. A properly designed and implemented switch should be capable of receiving a packet and switching the packet to an appropriate output port at what is referred to wirespeed or linespeed, which is the maximum speed capability of the particular network. Basic ethernet wirespeed is up to 10 megabits per second, Fast Ethernet is up to 100 megabits per second, and Gigabit Ethernet is capable of transmitting data over a network at a rate of up to 1,000 megabits per second. The newest Ethernet is referred to as 10 Gigabit Ethernet and is capable of transmitting data over a network at a rate of up to 10,000 megabits per second. As speed has increased, design constraints and design requirements have become more and more complex with respect to following appropriate design and protocol rules and providing a low cost, commercially viable solution. For example, high speed switching requires high speed memory to provide appropriate buffering of packet data; conventional Dynamic Random Access Memory (DRAM) is relatively slow, and requires hardware-driven refresh. The speed of DRAMs, therefore, as buffer memory in network switching, results in valuable time being lost, and it becomes almost impossible to operate the switch or the network at linespeed. Furthermore, external CPU involvement should be avoided, since CPU involvement also makes it almost impossible to operate the switch at linespeed. Additionally, as network switches have become more and more complicated with respect to requiring rules tables and memory control, a complex multi-chip solution is necessary which requires logic circuitry, sometimes referred to as glue logic circuitry, to enable the various chips to communicate with each other. Additionally, cost/benefit tradeoffs are necessary with respect to expensive but fast SRAMs versus inexpensive but slow DRAMs. Additionally, DRAMs, by virtue of their dynamic nature, require refreshing of the memory contents in order to prevent losses thereof. SRAMs do not suffer from the refresh requirement, and have reduced operational overhead which compared to DRAMs such as elimination of page misses, etc. Although DRAMs have adequate speed when accessing locations on the same page, speed is reduced when other pages must be accessed.

[0005] Referring to the OSI 7-layer reference model discussed previously, the higher layers typically have more information. Various types of products are available for performing switching-related functions at various levels of the OSI model. Hubs or repeaters operate at layer one, and essentially copy and "broadcast" incoming data to a plurality of spokes of the hub. Layer two switching-related devices are typically referred to as multiport bridges, and are capable of bridging two separate networks. Bridges can build a table of forwarding rules based upon which MAC (media access controller) addresses exist on which ports of the bridge, and pass packets which are destined for an address which is located on an opposite side of the bridge. Bridges typically utilize what is known as the "spanning tree" algorithm to eliminate potential data loops; a data loop is a situation wherein a packet endlessly loops in a network looking for a particular address. The spanning tree algorithm defines a protocol for preventing data loops. Layer three switches, sometimes referred to as routers, can forward packets based upon the destination network address. Layer three switches are capable of learning addresses and maintaining tables thereof which correspond to port mappings. Processing speed for layer three switches can be improved by utilizing specialized high performance hardware, and off loading the host CPU so that instruction decisions do not delay packet forwarding.

SUMMARY OF THE INVENTION:

[0006] The invention is directed to a switch using indicators for address learning. The switch has a first activator configured to control a first indicator to indicate when a source address needs to be learned and when a source address has been learned; a second activator configured to control a second indicator to indicate when a destination address has not been learned and when a destination address has been learned; and a third activator configured to control a third indicator to indicate when a source address has not been learned in all switches.

[0007] In another embodiment the invention is directed to a switch for address learning. The switch has a first indicator means for indicating when a source address needs to be learned and when a source address has been learned; a second indicator means for indicating when a destination address has not been learned and when a destination address has been learned; and a third indicator means for indicating when a source address has not been learned in all switches.

[0008] The invention is also directed to a method of address learning. The method has the steps of receiving a packet in a switch and performing a destination address lookup. An first indicator is set if there is a destination address lookup failure. A second indicator is set if the first indicator was set before the destination address lookup, there is no destination lookup failure, and the switch is the originating switch. A source address lookup is performed and third indicator is set if there is a source address lookup failure or if an L_State Bit is set. An address is learned if the third indicator is set.

BRIEF DESCRIPTION OF THE DRAWINGS:

[0009] The objects and features of the invention will be more readily understood with reference to the following description and the attached drawings, wherein:

Figure 1 is an illustration of an eight port gigabit switch having a 10 gigabit Ethernet interface;

Figure 2 is an illustration of eight gigabit switches connected to one another through a fabric interface;

Figure 3 is an illustration three modules connected to one another through a fabric interface;

Figure 4A is an illustration of a module header format

Figure 4B is an illustration of an extended module header format

Figure 5 is an illustration of an ARL Table; and

Figure 6 is a flow diagram of address learning.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS:

[0010] FIG. 1 is an illustration of a port gigabit switch (pGS) 100. In this example, pGS 100 has eight gigabit ports 102 and a 10-GE (Gigabit Ethernet) Interface 104, but it will be understood that other configurations are within the scope of the invention.

[0011] pGS 100 is a Layer 2 device with on-chip Media Access Control (MAC) address tables and a Fast Filtering Processor (FFP). The device can operate in standalone mode or can be connected to other devices including other pGS devices. pGS 100 is a single chip non-blocking switching solution for eight gigabit Ethernet ports and a 10-GE link. The FFP provides accelerated packet forwarding and enhanced packet flow through the use of functions such as classification, metering, shaping and re-marking.

[0012] Each gigabit port 102 of pGS 100 is associated with a corresponding MAC having an ingress for receiving data packets and an egress for transmitting data packets. Gigabit ports 102 support 10/100/1000 autonegotiation where the MACs operating at 10/100 Mbps support both half and full duplex operation and only full duplex operation at 1000 Mbps.

[0013] In one embodiment of the invention each of the gigabit ports 102 can be connected to an internal communications bus, such as a CPS channel. The CPS channel is actually three separate channels, referred to as a C-channel, a P-channel and a S-channel. The C-channel can be 128 bits wide, and can run at 132 MHz. Packet transfers between ports occur on the C-channel. Since this channel is used for data transfer, there is no overhead associated with its use. The P-channel or protocol channel is synchronous or locked with the C-channel. During data transfers, the message header is sent via the P-channel. The P-channel can be 32 bits wide and can run at 132 MHz. The S or sideband channel runs at 132 MHz, and is 32 bits wide. The S-channel is used for functions such as for conveying Port Link Status, receive port full, port statistics, ARL table synchronization, memory and register access to CPU and other CPU management functions, and global memory full and common memory full notification.

[0014] Arbitration for the CPS channel occurs out of band. The CPS channel is monitored by each of the gigabit ports 102, and matching destination ports respond to appropriate transactions. C-channel arbitration is a demand priority round robin arbitration mechanism. If no requests are active, however, the default module, which can be selected during the configuration of pGS 100 can park on the channel and have complete access thereto. If all requests are active, the configuration of pGS 100 is such that a Pipeline Memory Management Unit (PMMU) is granted access every other clock cycle, and gigabit ports 102 share equal access to the C-channel on a round robin basis. The PMMU and gigabit ports 102 alternate access, and the PMMU gains access every other cycle.

[0015] A plurality of messages can be placed on the protocol P-channel in order to properly direct flow of data flowing on the C-channel. Since the P-channel can be typically 32 bits wide, and a message typically requires 128 bits, four smaller 32 bit messages are put together in order to form a complete P-channel message. The following list identifies some of the fields and functions and the various bit counts of the 128 bit message on the P-channel. These are explanatory in nature and not intended to be limiting.

[0016] Opcode - 2 bits long - Identifies the type of message present on the C channel 81;

[0017] IP Bit - 1 bit long - This bit is set to indicate that the packet is an IP switched packet;

[0018] IPX Bit - 1 bit long - This bit is set to indicate that the packet is an IPX switched packet;

[0019] Next Cell - 2 bits long - A series of values to identify the valid bytes in the corresponding cell on the C channel 81;

[0020] SRC DEST Port - 6 bits long - Defines the port number which sends the message or receives the message, with the interpretation of the source or destination depending upon Opcode;

[0021] Cos - 3 bits long - Defines class of service for the current packet being processed;

[0022] J - 1 bit long - Describes whether the current packet is a jumbo packet;

[0023] S - 1 bit long - Indicates whether the current cell is the first cell of the packet;

[0024] E - 1 bit long - Indicates whether the current cell is the last cell of the packet;

[0025] CRC - 2 bits long - Indicates whether a Cyclical Redundancy Check (CRC) value should be appended to the packet and whether a CRC value should be regenerated;

[0026] P Bit - 1 bit long - Determines whether MMU should Purge the entire packet;

[0027] Len - 7 bytes - Identifies the valid number of bytes in current transfer;

[0028] - 2 bits - Defines an optimization for processing by the CPU 52; and

[0029] Bc/Mc Bitmap - 28 bits - Defines the broadcast or multicast bitmap. Identifies egress ports to which the packet should be set, regarding multicast and broadcast messages.

[0030] Untag Bits/Source Port - 28/5 bits long - Depending upon Opcode, the packet is transferred from Port to MMU, and this field is interpreted as the untagged bit map. A different Opcode selection indicates that the packet is being transferred from MMU to egress port, and the last six bits of this field is interpreted as the Source Port field. The untagged bits identifies the egress ports which will strip the tag header, and the source port bits identifies the port number upon which the packet has entered the switch;

[0031] U Bit - 1 bit long - For a particular Opcode selection (0x01, this bit being set indicates that the packet should leave the port as Untagged; in this case, tag stripping is performed by the appropriate MAC;

[0032] CPU Opcode - 18 bits long - These bits are set if the packet is being sent to the CPU for any reason. Opcodes are defined based upon filter match, learn bits being set, routing bits, destination lookup failure (DLF), station movement, etc;

[0033] Time Stamp - 14 bits - The system puts a time stamp in this field when the packet arrives, with a granularity of 1 μsec.

[0034] The opcode field of the P-channel message defines the type of message currently being sent. While the opcode is currently shown as having a width of 2 bits, the opcode field can be widened as desired to account for new types of messages as may be defined in the future.

[0035] An early termination message is used to indicate that the current packet is to be terminated. During operation, as discussed in more detail below, the status bit (S) field in the message is set to indicate the desire to purge the current packet from memory. Also in response to the status bit all applicable egress ports would purge the current packet prior to transmission.

[0036] The Src Dest Port field of the P-channel message, as stated above, define the destination and source port addresses, respectively. Each field is 6 bits wide and therefore allows for the addressing of sixty-four ports.

[0037] The CRC field of the message is two bits wide and defines CRC actions. Bit 0 of the field provides an indication whether the associated egress port should append a CRC to the current packet. An egress port would append a CRC to the current packet when bit 0 of the CRC field is set to a logical one. Bit 1 of the CRC field provides an indication whether the associated egress port should regenerate a CRC for the current packet. An egress port would regenerate a CRC when bit 1 of the CRC field is set to a logical one. The CRC field is only valid for the last cell transmitted as defined by the E bit field of P-channel message set to a logical one.

[0038] As with the CRC field, the status bit field (st), the Len field, and the Cell Count field of the message are only valid for the last cell of a packet being transmitted as defined by the E bit field of the message.

[0039] Last, the time stamp field of the message has a resolution of 1 μs and is valid only for the first cell of the packet defined by the S bit field of the message. A cell is defined as the first cell of a received packet when the S bit field of the message is set to a logical one value.

[0040] As is described in more detail below, the C channel and the P channel are synchronously tied together such that data on C channel is transmitted over the CPS channel while a corresponding P channel message is simultaneously transmitted.

[0041] The S channel is a 32-bit wide channel which provides a separate communication path within the pGS 100. The S channel is used for management by a Central Processing Unit (CPU), pGS 100 internal flow control, and pGS 100 inter-module messaging. The S channel is a sideband channel of the CPS channel, and is electrically and physically isolated from the C channel and the P channel. It is important to note that since the S channel is separate and distinct from the C channel and the P channel, operation of the S channel can continue without performance degradation related to the C channel and P channel operation. Conversely, since the C channel is not used for the transmission of system messages, but rather only data, there is no overhead associated with the C channel and, thus, the C channel is able to free-run as needed to handle incoming and outgoing packet information.

[0042] The S channel of the CPS channel provides a system wide communication path for transmitting system messages throughout pGS 100.

[0043] The message format for an S channel message on an S channel can be formed of four 32-bit words. The bits of the fields of the words are defined as follows:

[0044] Opcode - 6 bits long - Identifies the type of message present on the S channel;

[0045] Dest Port - 6 bits long - Defines the port number to which the current S channel message is addressed;

[0046] Src Port -6 bits long - Defines the port number of which the current S channel message originated;

[0047] COS - 3 bits long - Defines the class of service associated with the current S channel message; and

[0048] C bit - 1 bit long - Logically defines whether the current S channel message is intended for the CPU 52.

[0049] Error Code - 2 bits long - Defines a valid error when the E bit is set;

[0050] DataLen - 7 bits long - Defines the total number of data bytes in the Data field;

[0051] E bit - 1 bit long - Logically indicates whether an error has occurred in the execution of the current command as defined by opcode;

[0052] Address - 32 bits long - Defines the memory address associated with the current command as defined in opcode;

[0053] Data - 0-127 bits long - Contains the data associated with the current opcode.

[0054] With the configuration of the CPS channel as explained above, the decoupling of the S channel from the C channel and the P channel is such that the bandwidth on the C channel can be preserved for cell transfer, and that overloading of the C channel does not affect communications on the sideband channel.

[0055] FIG. 2 is an is an illustration of an fGS fabric 200. An fGS fabric provides for interconnections among devices. In this example fGS fabric 200 has eight 10-GE (GigabitEthernet) interfaces 210. Each of the eight pGS switches 220 each have eight gigabit ports 230 and a 10-GE interface 240. Each of the eight pGS switches 220 are interconnected to one another through fGS fabric 200. Each of the 10-GE interfaces 240 of each of the pGS switches 220 are connected to an associated 10-GE interface 210 of fGS fabric 200. In this way the eight pGS switches 220 are interconnected through fGS fabric 200 by the 10-GE interfaces provided on each of the pGS switches 220 and the fGS fabric 200.

[0056] FIG. 3 is a block diagram of three modules or switches 310, 320 and 330. Each of the modules 310, 320 and 330 can be a pGS switch as illustrated in FIG. 2 having a plurality of ports used to transfer data packets. A fabric 340 connects each of the modules 310, 320 and 330 to one another as illustrated in FIG. 2 using 10-GE interfaces.

[0057] In the example illustrated in FIG. 3 an address A is associated with Port 1 of Module 1, Address B is associated with Port 1 of Module 2 and Address C is associated with Port 1 of Module 3. As discussed previously a packet has a source address and a destination address. If a packet is received in Port 1 of Module 2 and is destined for Address A, Module 2 will send the packet to Module 1 and Module 3 in order to determine which port is the correct port. Module 2 must send the packet to all modules since Module 2 has no information to determine which port on which module is used to transmit a packet to Address A.

[0058] In order to speed up processing it is advantageous to implement a way for each of the modules to know which ports on which modules are associated with specific addresses. This can be referred to address “learning”. Once a module learns an address the module can forward the packet directly to the proper port on the proper module without sending the packet to all modules and ports.

[0059] One way to learn an address is to send a copy of a packet to a Central Processing Unit (CPU). The CPU can in turn formulate a control packet and send the control packet to other modules. An acknowledgement based system can be implemented so that the sending module knows that a given address is learned on all modules. If an acknowledgement is not received from all modules, then the CPU can send the control packet to those modules from which the CPU did not receive an acknowledgement from. This can be accomplished through either hardware or software or a combination of both.

[0060] In one embodiment of the invention, in order to avoid the difficulties of acknowledgement based system such as complex implementation and difficulties in verifying in silicon, a system can be implemented so that the hardware provides assistance for learning in the ARL table in addition to the CPU is used to learn and verify address such as MAC addresses. The mechanism is simple to implement, less complex to design and will speed up the address learning process.

[0061] The basic system that can be used to communicate control information from one switch or module to another switch or module can be through the use of a module header. The module header can be appended to the front of a packet and the receiving module will decode the module header and take appropriate action. FIG. 4A is an illustration of a basic module header format and FIG. 4B is an illustration of an extended module header. The following list identifies the fields and functions and the various bit counts of a basic module header.

[0062] Opcode/C - 3 bits long - Identifies the packet type;

[0063] Src_Port/SRC_TGID - 6 bits long - Indicates if a port is a source port or a trunk port;

[0064] M Bit - 1 bit long - Identifies if MT Module ID is valid;

[0065] Module Id Bitmap - 32 bits long - Bitmap of all modules, which are supposed to receive a packet;

[0066] Egress Port - 6 bits long - Identifies the port number on a remote module, which is supposed to receive a packet;

[0067] PFM - 2 bits long - Port filtering mode for the source port;

[0068] Source Mod Id - 5 bits long - Identifies the source module Id of a packet;

[0069] Data - N bits long - Indicates the number of bytes in the incoming packet not including the CRC;

[0070] CRC Of (Module Header + Data) - 32 bits long - CRC value including the data and the Module Header;

[0071] F - 1 bit long - Indicates if a destination address has been learned;

[0072] L - 1 bit long - Indicates when a source address needs to be learned;

[0073] NB - 1 bit long - Indicates when a module header is generated by another vendor;

[0074] FTD - 1 bit long - Indicates when a device can drop a packet;

[0075] GRP - 2 bits long - Identifies whether in normal mode or group mode; and

[0076] EH0 - 1 bit long - Identifies if there is a basic header format.

[0077] The following is a list of fields in the extended module header format as depicted in FIG. 4B.

[0078] EH1 - 1 bit long - Indicates if there is an additional extended module header;

[0079] Len - 3 bits long - Indicates the length of the extended module header; and

[0080] FFP label - 24 bits long - An FFP label transmitted by source module.

[0081] The Opcode/C field of the Module Header is 3 bits long and identifies packet type. If the field has a value of 0, this value identifies the frame as a Control Frame which is sent to a Control Processor. If the field has a value of 1, the packet is a unicast packet and the egress port is identified by a unique Module ID Bitmap (only one bit will be set in this field) and the egress port number. If the field has a value of 2, there is a DLF for the packet address and therefore the packet must be sent to multiple ports on the same module or multiple ports on different modules. If the field has a value of 3, the packet is identified as a multicast packet. If the field has a value of 4, the packet is identified as broadcast and is destined to multiple ports on the same module or multiple ports on different modules. If the field has a value of 5, a mirroring function needs to be performed if the M bit is set in the module header. If the filed has a value of 6, this indicates that a packet is destined to the processor fabric. When this Opcode value is used the module id bitmap represents the module id of the fabric.

[0082] The source port of a packet is indicated when the most significant bit of the Src_Port/SRC_TGID field is set. If the most significant bit is not set the source port is a trunk group and the field value is the TGID of the trunk group.

[0083] When the M Bit is set mirroring has been activated and the MT Module ID becomes a valid field.

[0084] The F and L bits are used for address learning. When the F Bit is set it is used for address learning. When a packet is received in the destination module with the F Bit set, an S-Channel message to the ARL will be generated to clear the L_State bit associated with the destination MAC address of a packet. When the L Bit is set, an S-Channel message to the ARL will be generated to learn the source MAC address of a packet.

[0085] The FTD field is a First-To-Drop Bit that indicates that a receiving device can drop a packet if there is congestion.

[0086] The GRP field indicates which operating mode the device in. In mode 1 or normal mode all bits in the module id bitmap are interpreted as module id's. In mode 2 or group mode, the upper two bits of the module id are interpreted as a group number and the lower thirty bits are interpreted module id's.

[0087] If the EH0 bit is set to zero this indicates that it is in basic header module format. If the EH0 bit is set to 1 then this indicates that the is an extended module header.

[0088] The EH1 filed of the extended module header indicates whether there is an additional module header. If the EH1 bit is set then there is an additional module header. If the EH1 bit is not set then there is no additional module header.

[0089] The LEN field of the extended module header indicates the length of the extended module header. This can be done in two byte segments (i.e. granularity of 2 bytes). The maximum length of an extended module header can be sixteen bytes.

[0090] The FFP label can be used so that it can be picked up from a rules table upon a full match.

[0091] As discussed above the F and L Bits are provided in the header to assist in address learning. When a source module learns an address, an IPIC can set the L bit in the module header by generating an S-Channel message to the ARL logic to learn the source address of a packet. The packet will then be sent to the destination. If the destination address is found in the ARL table and destination port is resolved the F Bit is set to zero indicating that the destination address has been learned. Otherwise the F bit is set to one indicating a destination lookup failure in the source module.

[0092] When a packet is received in a destination module, if the L bit is set, then the source MAC address of the packet is learned along with the source port number or the trunk group id which is present in the module header. In addition if the F Bit is set, this indicated that the destination address was not found in the source module. In each module, a state information, L_State, is maintained for each address that is present in an ARL table as depicted in FIG. 5. When a source address is learned in a module it will set the L_State bit to zero, indicating that the packet was sent to all other modules (defined by VLAN) for address learning purposes. If a packet is received the next time with the F Bit set to one in the module header for that particular MAC address, the L_State Bit is set. If after a source lookup , the address is found but the L_State Bit is zero, the packet is sent to all the modules with the L Bit set in the module header as explained above.

[0093] The above mechanism in some cases will only be applicable for a unicast packet and for address learning. CPU manipulation in some cases can be expected for the following: 1) Address Delete, 2) Station Movement, 3) L3 Address Learning, 4) Multicast address learning and delete.

[0094] Address learning as discussed above can be implemented as follows. When an address is learned in the source module, the ingress will clear the L_State Bit in the ARL Table. This will also be indicated to the IPIC port by setting the LP Bit in the P-Channel. The IPIC logic will pick up the LP Bit from the P-Channel header and set that as the value for the L Bit in the module header before sending the packet to other modules.

[0095] When the packet arrives at an ingress port and if the address is already learned in the ARL Table, the LP Bit in the P-Channel will be set to zero. A DA lookup will be performed and if the VID+MAC address is not found in the table, the ingress logic will set the FP bit in the P-Channel. The IPIC logic will pick up the FP Bit from the P-Channel header and set that as the value for the F Bit in the module header before sending the packet to other modules.

[0096] When a packet arrives at the destination module or the ingress to the IPIC, if the L bit is set in the module header, the IPIC will generate an S-Channel message to learn the source address of the packet to the ARL logic. If the F Bit is set in the module header, the IPIC will generate an S-Channel message to the ARL logic. The ARL logic will set the L_State Bit for entries associated with the destination MAC address in the packet if that module is the owner of the MAC address.

[0097] Aging of the entries can be done by the owner and learned addresses will be eliminated if certain condition are met. For example an entry may be eliminated because of inactivity or non-use.

[0098] From the above it is evident that address learning can be accomplished through the use of three indicators. In one example the indicator can be bits or fields located in the module header and ARL tables. As discussed above, the L_State Bit in the ARL table and the L and F Bits in the module header are utilized for address learning. The L Bit is used to indicate when a source address needs to be learned. The F Bit is used to indicated when a destination address has been learned. The L_State Bit is used to indicate when a source address has not been learned.

[0099] The L Bit is set if the source address is not found in the ARL table and must be learned.

[0100] The F Bit is set when there is a destination lookup failure. Simply stated, the F Bit is set when the destination address is not found in the ARL table.

[0101] The L_State Bit is set if the F Bit is set before a destination lookup, there is no destination lookup failure and the address is at the originating switch or module.

[0102] In order to more full understand the operation of the invention and the use of the indicators as explained above several examples are described below with reference to FIG. 3. Stations A, B and C reside on port 1 of modules 1, 2 and 3 and all ports belong to the same VLAN, VID=1.

[0103] **EXAMPLE 1**

[0104] In this first example Station A in Module 1 sends a packet which is destined for Station B on port 1 in Module 2. Port 1 in Module 1 receives the packet. Since the source address A is not learned in the ARL Table of Module 1, the address is learned and the L Bit is set in the module header. A Destination Address (DA) lookup is then performed and since B is not present there is a Destination Address Lookup Failure (DLF). Since there is a DLF the F Bit is set, the L_State is set to zero and the packet will be sent to all modules. The status of the fields in the ARL Table, L Bit and F Bit for Module 1 are as follows.

MODULE 1

MAC Address	1
Module	1
VID	1
Port	1
L_State	0
L Bit	1
F Bit	1

[0105] The packet is then sent to Modules 2 and 3. When the packet is received in Module 2 the source address is learned since the L Bit is set. The L_State Bit is in a "don't care" state since module 2 is not the owner of the packet. A DA lookup is then performed resulting in a DLF. The F Bit remains set since there is a DLF. The status of the fields in the ARL Table, L Bit and F Bit for Module 2 are as follows.

MODULE 2

MAC Address	A
Module	1
VID	1
Port	1
L_State	X

L Bit	1
F Bit	1

[0106] When the packet is received in Module 3, the source is learned in Module 3 since the L Bit is set. The L_State Bit is in a “don’t care” state since Module 3 is not the owner. A DA is then performed resulting in a DLF. The F Bit remains set since there is a DLF. The status of the fields in the ARL Table, L Bit and F Bit for Module 3 are as follows.

MODULE 3

MAC Address	A
Module	1
VID	1
Port	1
L_State	X
L Bit	1
F Bit	1

[0107] **EXAMPLE 2**

[0108] This second example is performed after execution of the first example. Therefore the ARL Tables, L Bits and F Bits for each module are in a state as described with relation to Example 1. In this second example Station B on port 1 in Module 2 sends a packet to Station A on port 1 in Module 1. Initially, Station B sends a packet addressed to Station A. Port 1 in Module 2 receives the packet. Since the source address B is not learned in the ARL Table of Module 2, the address is learned and the L Bit is set. A DA lookup is then done and since MAC address A is present in the ARL table the F Bit is set to zero, L_State is set to zero and the packet is only sent to Module 1. The status of the fields in the ARL Table, L Bit and F Bit for Module 2 are as follows.

MODULE 2

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1
L_State	X	0
L Bit	1	1
F Bit	1	0

[0109] When the packet is received in Module 1 the source address B is learned since the L Bit is set. The L_State Bit is in a "don't care" state since it is not the owner. The module header indicates that the packet is to be sent to port 1. The status of the fields in the ARL Table, L Bit and F Bit for Module 1 are as follows.

MODULE 1

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1

L_State	0	X
L Bit	1	1
F Bit	1	0

[0110] **EXAMPLE 3**

[0111] This third example is independent of the first two examples. In this example Station A on port 1 in Module 1 sends a packet to Station B on port 1 in Module 2 where each of the modules has learned address A and address B has been learned on Modules 2 and 3 but not on Module 1. The ARL Tables, L Bits and F Bits for each module are as follows.

MODULE 1

MAC Address	A
Module	1
VID	1
Port	1
L_State	0
L Bit	0
F Bit	0

MODULE 2

MAC Address	A	B
Module	1	2

VID	1	1
Port	1	1
L_State	X	0
L Bit	0	0
F Bit	0	0

MODULE 3

MAC Address	A
Module	1
VID	1
Port	1
L_State	X
L Bit	0
F Bit	0

[0112] Initially, Station A sends a packet addressed to station B. Port 1 in Module 1 receives a packet. Module 1 does a DA lookup for address B and since address B is not present in the ARL Table there is a DLF. Therefore the F Bit is set and the L_State bit is set to zero. Since the source address is already learned in the ARL Table and the L_State is set to zero, the L Bit is not set. Since the destination address has not been learned the packet is sent to Modules 2 and 3. The status of the fields in the ARL Table, L Bit and F Bit for Module 1 are as follows.

MODULE 1

MAC Address	A
Module	1
VID	1
Port	1
L_State	0
L Bit	0
F Bit	1

[0113] When the packet is received in Module 2 a Source Address (SA) lookup is not performed since the L Bit is not set. However, a DA lookup is performed since the F Bit is set. Since address B has already been learned by Module 2, there is no DLF. Since the F Bit was set, there was no DLF and Module 2 is the owner the L_State Bit is set. The packet is then flooded to all ports in Module 2. The status of the fields in the ARL Table, L Bit and F Bit for Module 2 are as follows.

MODULE 2

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1
L_State	X	1
L Bit	0	0

F Bit	0	0

[0114] When the packet is received in Module 3, no SA lookup is performed since the L Bit is not set. Since the F Bit is set a DA lookup is done and the address is found since address B has been learned by Module 3. However since Module 3 is not the owner the L_State does not change. The status of the fields in the ARL Table, L Bit and F Bit for Module 3 are as follows.

MODULE 3

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1
L_State	X	X
L Bit	0	0
F Bit	0	0

[0115] EXAMPLE 4

[0116] This fourth example can be performed after execution of the third example. Therefore the ARL Tables, L Bits and F Bits for each module are in a state as described with relation to Example 3. In this fourth and last example Station B on port 1 of Module 2 sends a packet to Station A on port 1 of Module 1. When Module 2 receives the packet an SA lookup shows that the entry is present and the L_State is 1. Therefore the L Bit is set in the Module header. The L_State is then set to zero in the

ARL Table. A DA lookup on Module 2 for address A indicates that the packet is to be sent to Module 1. The status of the fields in the ARL Table, L Bit and F Bit for Module 2 are as follows.

MODULE 2

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1
L_State	X	0
L Bit	0	1
F Bit	0	0

[0117] When Module 1 receives packet 1 the source address B is learned in the ARL Table since the L Bit was set. The status of the fields in the ARL Table, L Bit and F Bit for Module 3 are as follows.

MODULE 1

MAC Address	A	B
Module	1	2
VID	1	1
Port	1	1
L_State	0	X
L Bit	0	1

F Bit	0	0

[0118] FIG. 6 is a flow diagram illustrating steps for implementing address learning. In step 600 an SA lookup is performed and the L_State bit are checked. If the there is an SLF or the L_State bit is set the L Bit is set in step 610. If there is no SLF or if the L_State bit is not set the L_Bit is set to zero in step 620.

[0119] In step 630 a DA lookup is performed. If there is no DLF the module checks to see if the DA was previously learned by a previous module in step 640. This can be checked by determining if the F Bit was set when received by the module or switch. If the DA was previously learned and this switch is the originating switch as is checked in step 650, the L_State Bit is set to 1 as illustrated in step 660. By setting the L_State bit to 1, this will indicate that the address still needs to be learned although most of the modules may have already learned the address.

[0120] In step 630 if there is a DLF the F Bit will be set in step 670 and will go onto further processing in step 690. Likewise, if the DA was not previously learned in step 640 or the present switch is not the originating switch the packet would go onto further processing in step 690.

[0121] The above-discussed configuration of the invention is, in a preferred embodiment, embodied on a semiconductor substrate, such as silicon, with appropriate semiconductor manufacturing techniques and based upon a circuit layout which would, based upon the embodiments discussed above, be apparent to those skilled in the art. A person of skill in the art with respect to semiconductor design and manufacturing would be able to implement the various modules, interfaces, and tables, buffers, etc. of the present invention onto a single semiconductor substrate, based upon the architectural description discussed above. It would also be within the scope of the invention to implement the disclosed elements of the invention in discrete electronic components, thereby taking advantage of the functional aspects of the invention without maximizing the advantages through the use of a single semiconductor substrate.

[0122] Although the invention has been described based upon these preferred embodiments, it would be apparent to those of skilled in the art that certain modifications, variations, and alternative constructions would be apparent, while remaining within the spirit and scope of the invention. In order to determine the metes and bounds of the invention, therefore, reference should be made to the appended claims.